Lab Assignment

Cybersecurity Professional Program

Introduction to Python for Security

Network Communication

PY-06-L4

Login

Lab Objective

Implement authentication and verify incoming connections.

Lab Mission

Create a login interface using sockets.

Lab Duration

20-30 minutes

Requirements

- Basic knowledge of Python.

Resources

- Environment & Tools
  - Windows, macOS, Linux
    - PyCharm
    - Python 3

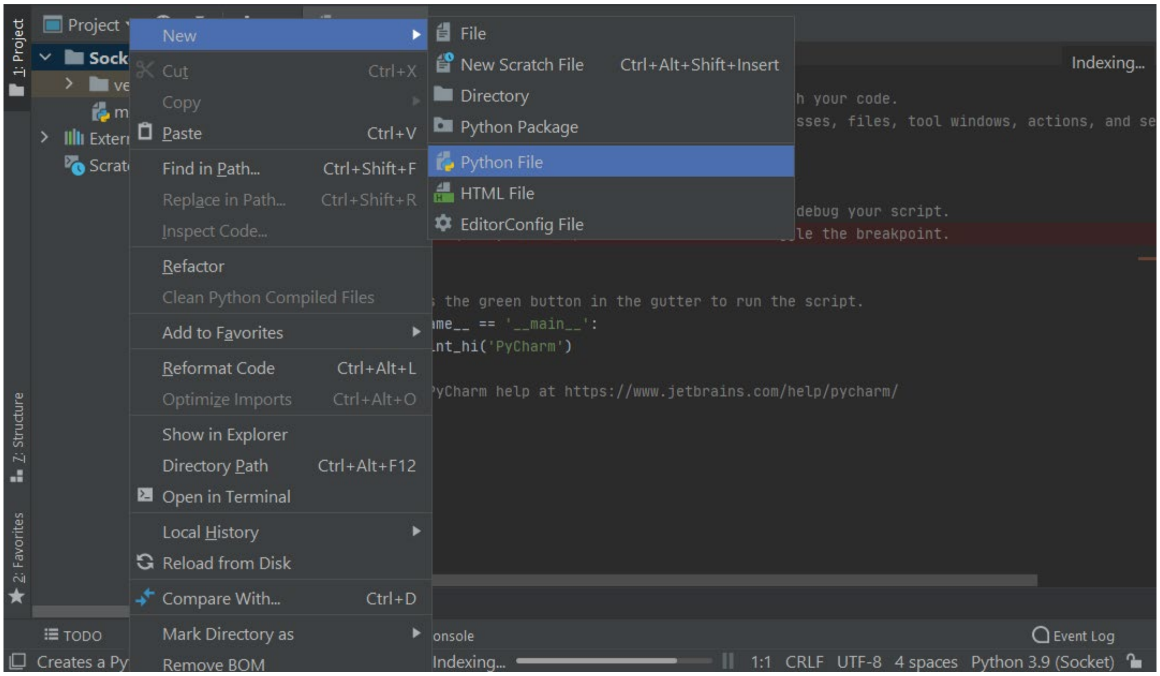
Textbook References

- Chapter 6: Network Communication
  - Section 4: Echo Communication

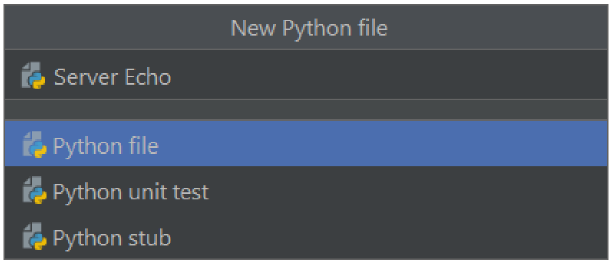
Lab Task 1: Server Socket Creation

In this task, you will create a server with a login dialog that asks a client to insert a username and password for authentication.

1 In the PyCharm project created in L1, create a new Python file by right-clicking the projects folder, and selecting **New → Python File**. Name the file **Server Echo**.



```
h your code.
sses, files, tool windows, actions, and se
debug your script.
le the breakpoint.
 the green button in the gutter to run the script.
ame__ == '__main__':
nt_hi('PyCharm')
yCharm help at https://www.jetbrains.com/help/pycharm/
```



Server Echo

2 Import the **socket** module to the file.

```
import socket
```

3 Create a socket variable.

```
import socket

s = socket.socket()
```

4 Bind the socket to accept connections from all IP addresses on port 4444.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))
```

5 Allow only one connection to the socket.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)
```

6 Accept a connection from clients and save the connection object and address to variables.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)

conn, addr = s.accept()
```

7 Send a welcome message to the connected client, and request for a username.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)

conn, addr = s.accept()

conn.send("Welcome to the server!\nPlease insert your Username:".encode())
```

8 Accept the username sent by the client and save it to a variable.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)

conn, addr = s.accept()

conn.send("Welcome to the server!\nPlease insert your Username:".encode())

username = conn.recv(2048).decode()
```

9 Request a password from the client.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))
s.listen(1)
conn, addr = s.accept()
conn.send("Welcome to the server!\nPlease insert your Username:".encode())
username = conn.recv(2048).decode()
conn.send("Please insert the Password:".encode())
```

10 Accept the password sent by the client and save it to a variable.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))
s.listen(1)
conn, addr = s.accept()
conn.send("Welcome to the server!\nPlease insert your Username:".encode())
username = conn.recv(2048).decode()
conn.send("Please insert the Password:".encode())
password = conn.recv(2048).decode()
```

11 Create a condition to check if the username is **John** and the password is **12345**.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)

conn, addr = s.accept()

conn.send("Welcome to the server!\nPlease insert your Username:".encode())

username = conn.recv(2048).decode()

conn.send("Please insert the Password:".encode())

password = conn.recv(2048).decode()

if username == "John" and password == "12345":
```

12 If the username and password are correct, send the client a welcome message.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))
s.listen(1)
conn, addr = s.accept()
conn.send("Welcome to the server!\nPlease insert your Username:".encode())
username = conn.recv(2048).decode()
conn.send("Please insert the Password:".encode())
password = conn.recv(2048).decode()
if username == "John" and password == "12345":
    conn.send(f"Welcome {username}".encode())
```

13 If the credentials are incorrect, send a message that tells the client that the password is wrong.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)

conn, addr = s.accept()

conn.send("Welcome to the server!\nPlease insert your Username:".encode())

username = conn.recv(2048).decode()

conn.send("Please insert the Password:".encode())

password = conn.recv(2048).decode()

if username == "John" and password == "12345":

    conn.send(f"Welcome {username}".encode())

else:
    conn.send("Wrong username or password".encode())
```

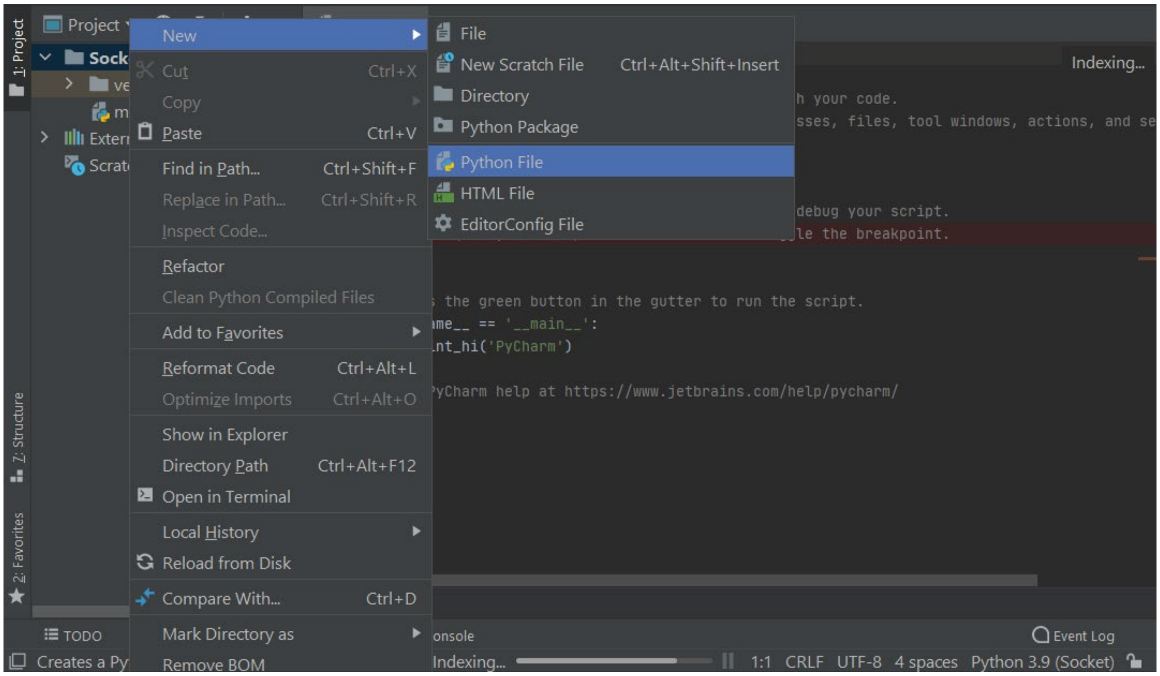
14 Close the connection.

```
import socket

s = socket.socket()

s.bind(("0.0.0.0", 4321))

s.listen(1)

conn, addr = s.accept()

conn.send("Welcome to the server!\nPlease insert your Username:".encode())

username = conn.recv(2048).decode()

conn.send("Please insert the Password:".encode())

password = conn.recv(2048).decode()

if username == "John" and password == "12345":

    conn.send(f"Welcome {username}".encode())

else:
    conn.send("Wrong username or password".encode())


s.close()
```

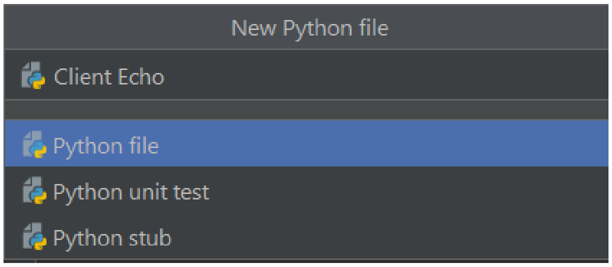
Lab Task 2: Client Socket Creation

In this task, you will create a client to connect to the server, and perform the authentication process.

1 In the PyCharm project created in L1, create a new Python file by right-clicking the projects folder, and selecting **New → Python File**. Name the file **Client Echo**.



```
h your code.
sses, files, tool windows, actions, and se
debug your script.
le the breakpoint.
 the green button in the gutter to run the script.
ame__ == '__main__':
nt_hi('PyCharm')
yCharm help at https://www.jetbrains.com/help/pycharm/
```



Client Echo

2 Import the **socket** module to the file.

```
import socket
```

3 Create a socket variable.

```
import socket

s = socket.socket()
```

4 Connect the socket to the listener in the local host.

```
import socket

s = socket.socket()

s.connect(("127.0.0.1", 4321))
```

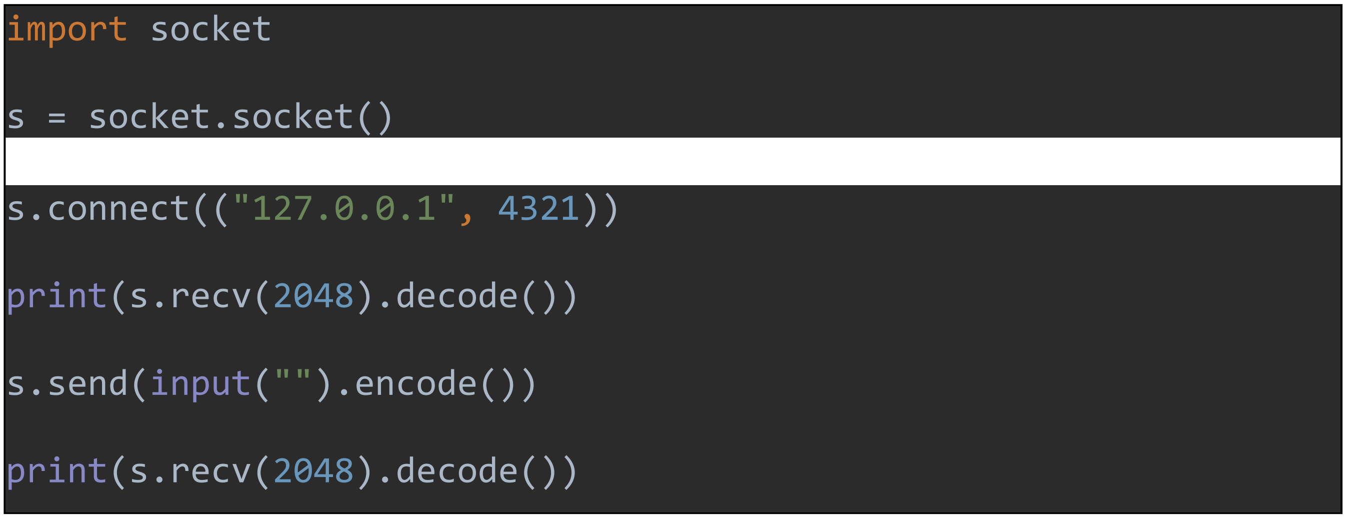
5 Print the first welcome message from the server.

```
import socket

s = socket.socket()

s.connect(("127.0.0.1", 4321))

print(s.recv(2048).decode())
```

6 Get an input for the username and send it to the server.

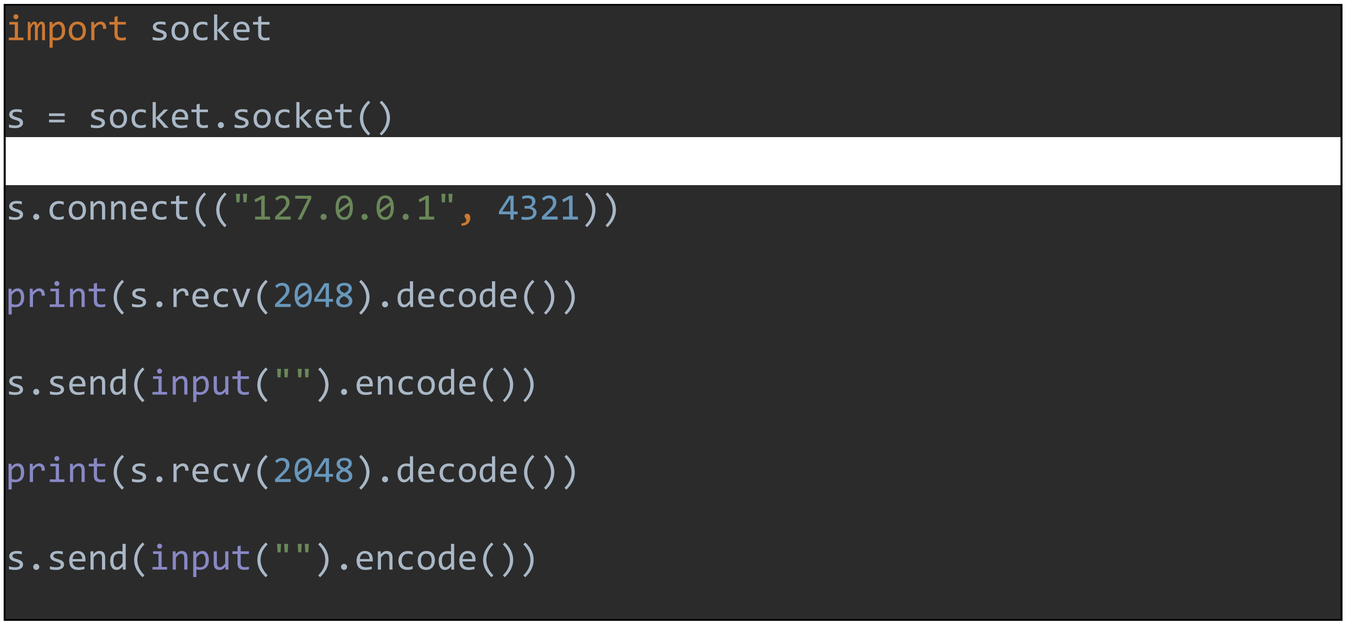
```
import socket

s = socket.socket()

s.connect(("127.0.0.1", 4321))

print(s.recv(2048).decode())

s.send(input("").encode())
```

7 Print the second message from the server that requests the password.



```python
import socket
s = socket.socket()
s.connect(("127.0.0.1", 4321))
print(s.recv(2048).decode())
s.send(input("").encode())
print(s.recv(2048).decode())
```

8 Get an input for the password and send it to the server.



```python
import socket
s = socket.socket()
s.connect(("127.0.0.1", 4321))
print(s.recv(2048).decode())
s.send(input("").encode())
print(s.recv(2048).decode())
s.send(input("").encode())
```

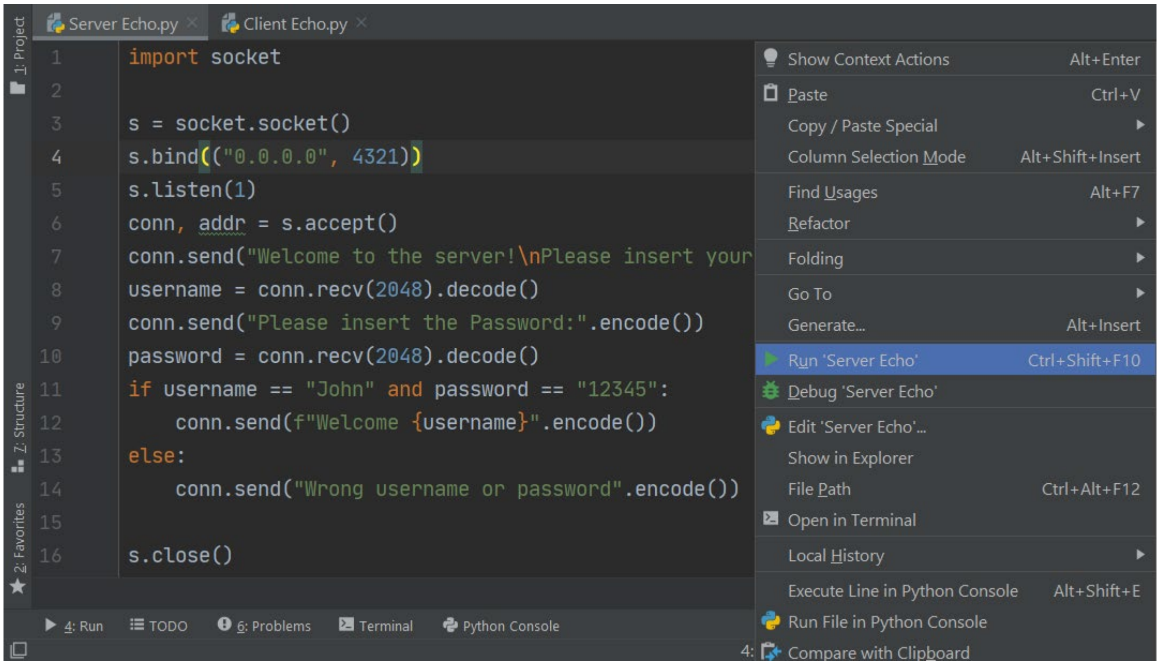
9 Print the final answer from the server regarding authentication.

```
import socket

s = socket.socket()

s.connect(("127.0.0.1", 4321))

print(s.recv(2048).decode())

s.send(input("").encode())

print(s.recv(2048).decode())

s.send(input("").encode())

print(s.recv(2048).decode())
```

10 Close the connection.

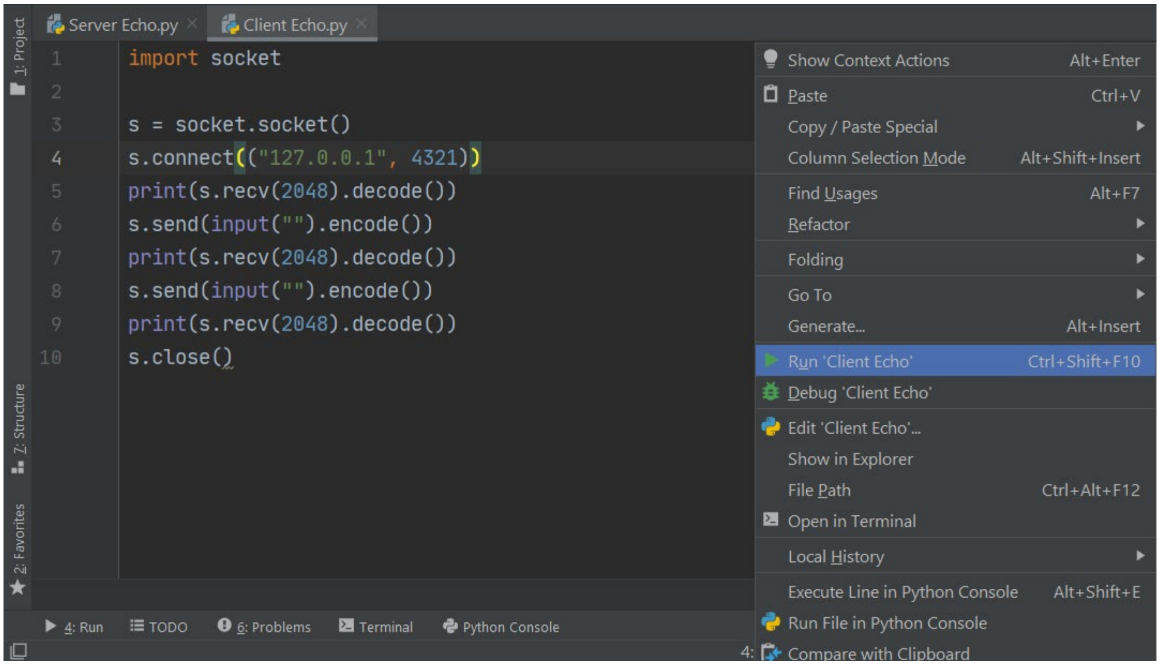
```
import socket

s = socket.socket()

s.connect(("127.0.0.1", 4321))

print(s.recv(2048).decode())

s.send(input("").encode())

print(s.recv(2048).decode())

s.send(input("").encode())

print(s.recv(2048).decode())

s.close()
```

11 Execute the server script.



```
import socket

s = socket.socket()
s.bind(("0.0.0.0", 4321))
s.listen(1)
conn, addr = s.accept()
conn.send("Welcome to the server!\nPlease insert your
username = conn.recv(2048).decode()
conn.send("Please insert the Password:".encode())
password = conn.recv(2048).decode()
if username == "John" and password == "12345":
    conn.send(f"Welcome {username}".encode())
else:
    conn.send("Wrong username or password".encode())

s.close()
```

12 Execute the client script.



```
import socket

s = socket.socket()
s.connect(("127.0.0.1", 4321))
print(s.recv(2048).decode())
s.send(input("").encode())
print(s.recv(2048).decode())
s.send(input("").encode())
print(s.recv(2048).decode())
s.close()
```